Course Design:
Modern Visual Recommender Systems

Kai Xin Thia
kaixin.thia@gatech.edu

Abstract**—** I have worked in the data industry for over seven years and had the privilege of designing, building, and deploying two recommender systems (RecSys) that went on to serve millions of customers across Southeast Asia. As technology evolved, so does RecSys. However, courses teaching RecSys are still very academic. I will develop an industry-focused, project-based, visual course to teach data scientists RecSys using modern tools.

1 INTRODUCTION

Recommender Systems (RecSys) are one of the most common data science models that we interact in our daily lives. From eCommerce sites like Amazon (Smith et al.,2017) to entertainment platforms like Netflix (Gomez-Uribe et al.,2015) to news portals (Beam, 2014) and more, there are always RecSys pushing new content that are "recommended for you". With the advancement in technology and techniques like deep learning, data scientists building RecSys must have the capability to employ new ways to improve their RecSys. Furthermore, data science is a multidisciplinary subject that combines math & statistics knowledge, hacking skills, and substantive domain expertise (Asamoah et al.,2018). It is not easy to develop a data scientist who is capable across such a broad spectrum of skills (Cleveland et al., 2014).

The problem is that there is a knowledge gap between what the school teaches and the reality students face when they take on roles as professional data scientists (Hardin et al.,2015). Students who are used to working in a small sandbox environment with toy datasets will be surprised by the lack of clean data, clear objectives, and the need to work on big data infrastructure in the real world.

Courses teaching data science are still very academic and lack the "**realistic and complex challenges that model real-world problems** faced in industrial settings" (Hopfgartner et al.,2020). Traditionally, students are introduced to simple data sets, such as the classic Iris flower data set (Fisher, 1936). While this data set allows students to build simple models and practice their theories, it is neither a realistic representation of what they will encounter at work nor challenging enough. The data is so small and straightforward that there is very little room for creativity.

Since there is significant variation across data science topics in terms of complexity, it can be **hard to design a pedagogy that is suitable for all branches of data science education**. It might be more feasible to focus on specific skill development or learning outcomes (Swamy, 2018). A common data science model like **RecSys**, with **data visualization** of the input and results (Kang et al., 2017), will also help students connect the dots between theory and application in the real world. Beyond just the technical implementation of RecSys, students can even debate and learn about the business considerations (He, R. et al., 2016), as there is no "correct answer" in the world of RecSys (Zhang et al., 2019). It is thus essential to develop a pedagogy that engages students, for example, how Emilio et al. developed three deep learning courses based on Kolb's four-stage experiential learning cycle.

The goal of developing good data scientists through a RecSys course is ambitious. It is, however, an important step to move the data science field forward, especially in a time when data scientists are going into industry instead of academia, and data literacy is becoming a common core (Dichev C. et al., 2017). The industry needs good content that can engage these students that might not come from traditional computer science, mathematics, or statistics background (Schuff, 2017). **My RecSys course should be suitable for students with a foundational understanding of data science and is comfortable with programming. Undergraduates, postgraduates, and data professionals** should be able to benefit from my content, as I will be exploring more challenging and cutting-edge content than typical RecSys 101 courses.

2 RELATED WORK

There is a growing trend of technical tutorials on platforms like Medium.com. Courses like "Simple Reinforcement Learning with Tensorflow" by Arthur Juliani (2016) is an excellent 10-part series that **covers a range of topics**, from the fundamentals of reinforcement learning to its modern approach. More importantly, **code notebooks are embedded within the content** for students to try it out; **this is an approach that I can emulate for my course.**

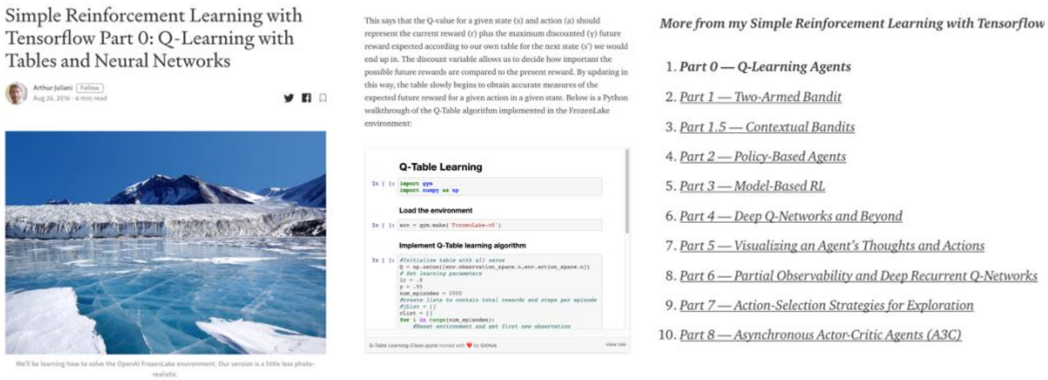


Figure 1 — Arthur's course on Medium (left), with embedded code notebooks (middle) and links to connect with the rest of the series (right)

On the recommender front, there are several authors on Medium who created content for RecSys. Several of them cover **only the introduction** of RecSys (Kordík, 2018; Huang, 2018; Rocca, 2019; Hui, 2020), in areas like collaborative filtering, singular value decomposition, and basic evaluation methods. Other authors **focus on specific topics** such as Tree-based Deep Model (Alibaba, 2018), deep generative models (Bacuet, 2019), deep learning in PyTorch (Gupta, 2018), and deep learning in Keras (Gutierrez, 2018). None of the RecSys courses are organized into a structured, in-depth course like Arthur's.

It seems that if students preferred a **structured** series of RecSys content, they would have to refer to more academic platforms like Coursera, which hosts an excellent RecSys specialization course (Konstan, 2020) or review the myriad of choices on Udemy (2020). Alternatively, Google (2020) has created a Recommendation Systems **Colab notebook** that guides students through the RecSys process with code. The content covered is very similar to the course outlined by Zheng, Y. (2019), who built a RecSys course targeting graduate students:

- Understand the objectives, tasks, and evaluations in recommender systems.
- Learn the necessary knowledge and skills in AI and data science to build recommendation models.
- Learn, implement, and develop classical approaches for traditional recommender systems with open-source libraries.
- Learn, implement, and develop novel approaches for traditional recommender systems with open-source libraries.

- Recommender systems for different applications

3 THE SOLUTION

I combine the **flexibility** of Medium, the **interactivity** of Colab, the **structure** of traditional courses with my **real-world deployment experience** to create a RecSys course that **dives a little deeper beyond the surface**. I want to contribute a new course that target students who hunger for more challenge and will like to learn about the modern approaches to RecSys.

3.1 Overview & Content Platform

For my course, I deliver it **using Medium.com and LinkedIn articles**, with links **to interactive code notebooks on a publicly accessible platform - Google Colab.** Students can execute the RecSys code on Colab without the need to install any additional software/packages/data sets. Since I have a lot of content to cover, **I plan to keep the materials to text and code only - I do not intend to include any video lectures.** Importantly, the course will be freely accessible by all. The course should be suitable for students with a foundational understanding of data science and is comfortable with programming.

3.2 Data

My primary data source is DeepFashion (Liu, Z. et al., 2016) from The Chinese University of Hong Kong, with 280K labeled images and is free for non-commercial use. This data set is large enough and has all the attributes I need.

Figure 2 — Images from DeepFashion paper (Liu, Z. et al., 2016)

For introductory materials, I also made use of fashion-mnist (Xiao, H. et al., 2017), as it is a smaller dataset that is easier to visualize and explore.

3.3 Course outline

The course begins with an **overview** of the subject and the business context. The materials start with a simple case study of a **basic recommender** before diving deeper into core models like **Convolutional Neural Networks (CNN)** and advanced considerations. The understanding of fashion in combination **across multiple outfits and temporal models** will be explored in the later stages before I end off with the latest developments in the field. Throughout the course, **reflections** by students will form part of the learning process; details are covered under the pedagogy section.

- How does a recommender work?
  - RecSys 101 – overview, definitions, examples
  - Reflection
  - Additional resources / course recommendation
- How to design a recommender?
  - The RecSys framework – walk through how to define the business problem, start small, scale, test & deploy
  - Metrics – accuracy, diversity, privacy, coverage, serendipity
  - Cold start problem
  - Reflection
  - Additional resources / course recommendation
- Intro to visual RecSys
  - Implement a similarity model on Fashion MNIST
  - How it works – overview of image embeddings
  - Visual RecSys – the visual problem, unique challenges, tools for visualizations, sample output
  - Reflection
  - Additional resources / course recommendation
- Intro to Convolutional Neural Networks (CNN)
  - Basic deep learning (DL) visual recommender - Convolutional Neural Networks (CNN)
  - How it works – CNN
  - Reflection

    - Additional resources / course recommendation
- Building a Personalized Real-Time Fashion Collection Recommender
    - Visual understanding of fashion items with deep learning
    - Generation of fashion wardrobe
    - How it works – recommendation with understanding
    - Reflection
    - Additional resources / course recommendation
- Temporal modeling of fashion
    - How it works – time decay and style evolution
    - Reflection
    - Additional resources / course recommendation
- Conclusion and next steps
    - The latest developments in RecSys
    - Key takeaways
    - Reflection
    - Additional resources / course recommendation

3.4 Pedagogy

Experiential learning and reflective learning will be a vital part of the pedagogy. Both experiential learning and reflective learning are especially powerful in handling complex materials (Moon, 2013) from a field such as data science.

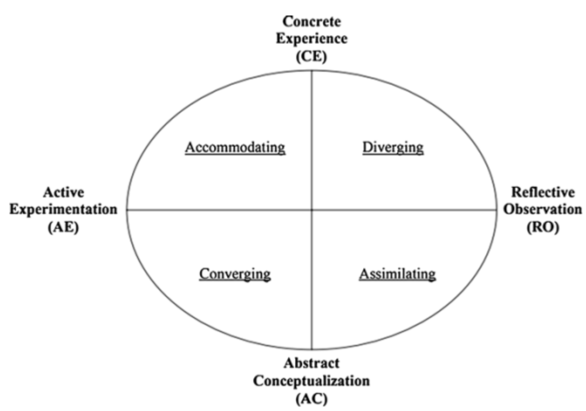


Figure 3 — The Experiential Learning Cycle and Basic Learning Styles (Kolb, 1984).

Many papers mentioned Kolb's model (Figure 2) and ways to integrate the model into the classroom. Kolb's himself evaluated the model in his updated paper: "According to the four-stage learning cycle depicted in Figure 1, immediate or concrete experiences are the basis for observations and **reflections**. These reflections are assimilated and distilled into abstract concepts from which new implications for action can be drawn. These implications can be actively tested and serve as guides in creating new experiences."(Kolb, 2014). An extension to learning about learning (meta-cognitive learning skills) (Kolb, 2009) and learning space (Kolb, 2005) can also be made. For example, "concrete experiences can be evoked by recalling past experiences, through **role-play**, or via **case studies**; reflective observation is cultivated by group discussions, reflective papers, and journals; abstract conceptualization is stimulated by lectures, print sources, and films; and active experimentation is often encouraged by means of problem-solving exercises such as mock proposals or role plays" (Lewis et al., 1994).

Kolb's theories have been implemented in courses by Emilio et al. (2017), who developed three deep learning courses [an advanced field in data science] **based on Kolb's experiential learning cycle**. The context is that "the learner has to be **actively engaged** in posing questions, investigating, experimenting, being curious, solving problems, assuming responsibility, being creative, and constructing meaning...acquiring skills requires more than 'monkey see, monkey do'". I plan to **integrate Kolb's cycle into my course** by encouraging active experimentation with interactive Colab code notebooks, concrete experience with case studies, reflective observation with short reflection essays, and abstract conceptualization with the course content.

According to Kross et al. (2019), "many people who currently teach data science are practitioners such as computational researchers in academia or data scientists in [the] industry". **As a practitioner-instructor myself** who works as a senior data scientist and with experience conducting training workshops, there are contributions that I can bring to the table. I can connect with students by sharing real-world experiences, inviting them to communities of practice, and teaching them "technical workflows that integrate authentic practices surrounding code, data, and communication" (Kross et al., 2019). Concretely, **industry case studies** might be a suitable medium to involve the students, **many who come with varying degrees of experience.** The case studies can be "a series of small problems in stages through which the students go on their own… (where) the focus is thinking about business analytics and verbalizing insights from analysis for presentation to a business and a technical audience" (Pachamanova, 2015).

One way to let students appreciate the complexity is to include a challenging real-world problem, as demonstrated in the paper by Lommatzsch et al. (2017). NewsREEL is a Recommender System (RecSys) challenge developed by the instructors and "requires participants to conduct data-driven experiments in NewsREEL Replay as well as deploy their best models into NewsREEL Live's 'living lab'" (Lommatzsch et al., 2017). Furthermore, Saltz et al. (2015) also emphasized the importance of a **project-focused** introduction to big data science courses, where participants have to solve a **real-world problem for a real-world client. For the course, I will design case studies that can connect the data, business, and technical problems.**

4 METHODOLOGY

Studies have shown that Massively Open Online Courses (MOOCs) have median completion rates of around **12.6%** (Jordan, 2015), with longer courses having lower completion rates. While it is hard to track the completion rate for a course like mine that is spread over a series of articles, platforms like Medium and LinkedIn do provide statistics on the number of views, time spent, and interactions with the content (likes, shares, etc.). **Thus, I will be evaluating my posts across both Medium and LinkedIn**, as the two platforms have different metrics and target audiences. Concretely, I will be collecting the following data for each Medium and LinkedIn posts, with **Medium's read ratio as an approximate comparison with MOOC competition rate.**

Medium

- Views **(Target: >100)**
- Reads
- Read Ratio **(Target: >15%)**
- Fans
- Average Reading Time in Mins

LinkedIn

- Views **(Target: >100)**
- Reshares
- Reactions **(Target: >15)**
- Comments

5 THE RESULTS

I posted the articles to Medium and LinkedIn every week over two months. The articles covered a **variety of case studies**, including RecSys at Spotify, Amazon, Netflix. I also included **applications of RecSys** in categorizing **COVID-19 x-ray** scans and how it can be applied **in real-time recommendations** with **hands-on code examples on Colab**. I submitted my articles on Medium to publishers like Analytics Vidhya (2020) and Towards Data Science (2020) to help increase the outreach of my articles. Overall, we have a standout piece, **"Building a Personalized Real-Time Fashion Collection Recommender" on Medium with over 1K views** and a standout piece on **LinkedIn "How does a Recommender Work?" with over 600 views**. It seems that we have different kinds of readers on the two platforms. **Medium readers prefer more in-depth data science articles**, with the more technical pieces on Convolutional Neural Networks (CNN) and their applications in COVID-19 scans, personalized recommendation, and temporal usage gaining more traction. In contrast, **LinkedIn readers flock towards the first three introductory articles** and the COVID-19 case study but do not seem that interested in the more technical pieces on CNN.

Furthermore, the read ratio of all the Medium articles is **all above 15%,** which is great, though the **average reading time is a minute or less**, suggesting that most readers are just **glancing through the materials**. I have already tried to keep the articles short and to the point, with 4min – 10min readings, excluding the code.

Table 1 — Results from Medium Articles

Article (with Links)	Views	Reads	Read Ratio	Fans	Total Length (Minutes to Read Article)	Average Reading Time (Minutes)	Publisher
How does a recommender work?	94	22	23%	1	6	0.63	AI in Plain English
How to Design a Recommender?	60	25	42%	4	7	1.08	Analytics Vidhya
Intro to Visual RecSys	40	12	30%	1	5	0.53	Analytics Vidhya

Article (with Links)	Views	Reads	Read Ratio	Fans	Total Length (Minutes to Read Article)	Average Reading Time (Minutes)	Publisher
Convolutional Neural Networks Recommender	55	17	31%	3	8	0.78	Analytics Vidhya
COVID-19 Case Study with CNN	109	53	49%	1	4	0.7	Analytics Vidhya
Building a Personalized Real-Time Fashion Collection Recommender	**1100**	**270**	**24%**	**20**	**10**	**1.01**	**Towards Data Science**
Temporal Fashion Recommender	222	90	41%	7	6	0.53	Towards Data Science
The Future of Visual Recommender Systems: Four Practical State-Of-The-Art Techniques	349	153	44%	6	7	0.58	Towards Data Science

Table 2 — Results from LinkedIn Articles

Article (with Links)	Views	Reshares	Reactions	Comments
How does a Recommender Work?	**643**	**5**	**118**	**5**
How to Design a Recommender?	274	2	32	0
Intro to Visual RecSys	133	1	10	0
Convolutional Neural Networks Recommender	75	0	17	0
COVID-19 Case Study with CNN	188	3	35	1
Building a Personalized Real-Time Fashion Collection Recommender	89	0	29	0
Temporal Modeling	46	0	16	0

Article (with Links)	Views	Reshares	Reactions	Comments
The Future of Visual Recommender Systems: Four Practical State-Of-The-Art Techniques	35	0	18	2

On the LinkedIn side, most of the reactions are "likes" with some words of encouragement under the comments; there are minimal interactions and feedback from the readers regarding the article. In terms of "views", LinkedIn has a lower average of 185 compared to the 253 of Medium, but Medium's views are highly skewed due to the one article with 1100+ views that was promoted by Medium and publisher Towards Data Science.

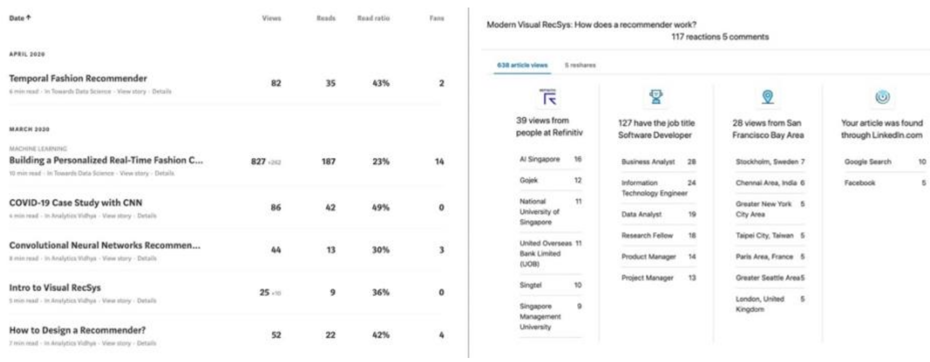


Figure 4 — Article Stats. Left: Medium. Right: LinkedIn.

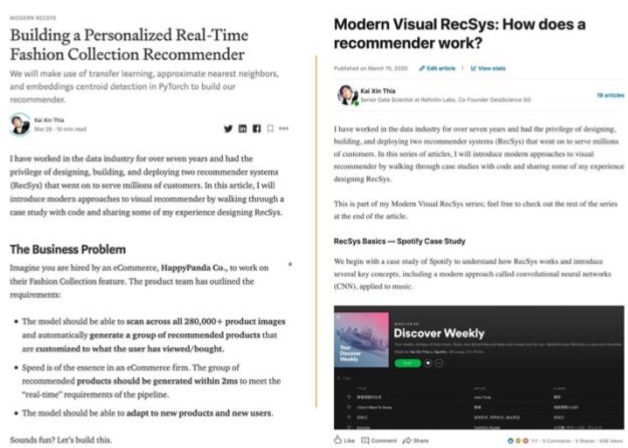
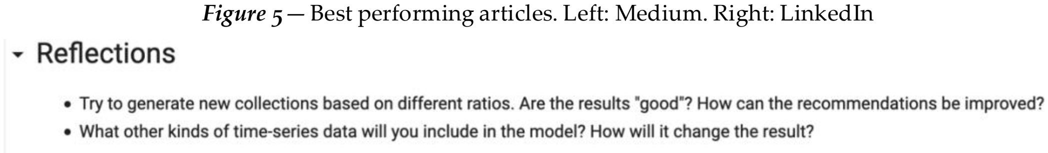


Figure 5 — Best performing articles. Left: Medium. Right: LinkedIn



Reflections

- Try to generate new collections based on different ratios. Are the results "good"? How can the recommendations be improved?
- What other kinds of time-series data will you include in the model? How will it change the result?

Figure 6 — Reflection questions at the end of each Colab notebook to encourage students to think deeper about the topic.

6 LIMITATIONS

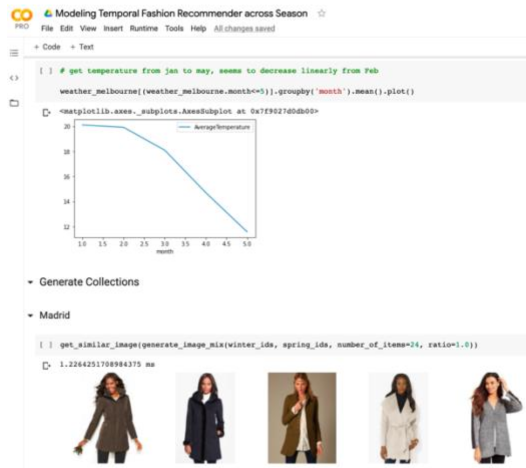


Figure 7— Colab interactive environment for learners to execute code and see the results immediately.

Despite the user metrics we collected, it is difficult to evaluate the course thoroughly. In particular, the interactive code environment, Colab, does not have any usage statistics, making it hard to measure its success. Moreover, there is no feedback loop from users regarding the course content (Is it relevant? Too easy? Too hard?) nor means to gather suggestions to improve the course.

Another problem is outreach. I just started my Medium account and only accumulated 20+ followers; this means that few people will read my articles unless Medium help to promote them.

7 CONCLUSION & FUTURE WORK

I managed to create a course that fits my vision of **bringing practical, case study-based materials into the hands of data science practitioners**. I even included a case study of **COVID-19 scans, to showcase the power and flexibility of the course materials**. The course received over a thousand views in two months and is unique in its content of **incorporating modern RecSys techniques like CNN with executable code examples tied to real-world problems and data sets**. There are areas of improvement, especially in the **feedback and evaluation cycles of the course**. Potential extensions of the project include live workshops, where I can walk through the course and observe how students interact with the materials. I can also conduct surveys and focus groups to obtain qualitative feedback on course improvements and validating if we are meeting the learning objectives.

8 REFERENCES

1. Abdollahi, B., & Nasraoui, O. (2016). Explainable Restricted Boltzmann Machines for Collaborative Filtering. ArXiv:1606.07129 [Cs, Stat]. http://arxiv.org/abs/1606.07129
2. Alibaba, T. (2018). Improving Recommender System with Tree-based Deep Model. https://medium.com/@alitech_2017/improving-recommender-system-with-tree-based-deep-model-e69f4a32d014
3. Analytics Vidhya (2020). https://medium.com/analytics-vidhya
4. Arthur, J. (2016). Simple Reinforcement Learning with Tensorflow Part 0: Q-Learning with Tables and Neural Networks. https://medium.com/emergent-future/simple-reinforcement-learning-with-tensorflow-part-0-q-learning-with-tables-and-neural-networks-d195264329d0
5. Asamoah, D. A., Doran, D., & Schiller, S. (2018). Interdisciplinarity in Data Science Pedagogy: A Foundational Design. Journal of Computer Information Systems, 1–8. https://doi.org/10.1080/08874417.2018.1496803
6. Bacuet, Q. (2019) Implementation of deep generative models for recommender systems in Tensorflow. https://medium.com/snipfeed/how-to-implement-deep-generative-models-for-recommender-systems-29110be8971f
7. Beam, M. A. (2014). Automating the News: How Personalized News Recommender System Design Choices Impact News Reception. Communication Research, 41(8), 1019–1041. https://doi.org/10.1177/0093650213497979
8. Cleveland, W. S. (2014). Data science: An action plan for expanding the technical areas of the field of statistics: Technical Areas of the Field of Statistics. Statistical Analysis and Data Mining: The ASA Data Science Journal, 7(6), 414–417. https://doi.org/10.1002/sam.11239
9. Dichev, C., & Dicheva, D. (2017). Towards Data Science Literacy. Procedia Computer Science, 108, 2151–2160. https://doi.org/10.1016/j.procs.2017.05.240
10. Emilio, S., Martin, M., Daniel, M., & Luis, B. (2017). Experiential Learning in Data Science: From the Dataset Repository to the Platform of Experiences. Ambient Intelligence and Smart Environments, 122–130. https://doi.org/10.3233/978-1-61499-796-2-122.
11. Fisher, R. A. (1936). THE USE OF MULTIPLE MEASUREMENTS IN TAXONOMIC PROBLEMS. Annals of Eugenics, 7(2), 179–188. https://doi.org/10.1111/j.1469-1809.1936.tb02137.x

12. Google, (2020). Recommendation Systems with TensorFlow. https://colab.research.google.com/github/google/eng-edu/blob/master/ml/recommendation-systems/recommendation-systems.ipynb
13. Gomez-Uribe, C. A., & Hunt, N. (2015). The Netflix Recommender System: Algorithms, Business Value, and Innovation. ACM Transactions on Management Information Systems, 6(4), 1–19. https://doi.org/10.1145/2843948
14. Gutierrez, P. (2018). DEEP BEERS: Playing with Deep Recommendation Engines Using Keras. https://medium.com/data-from-the-trenches/deep-beers-playing-with-deep-recommendation-engines-using-keras-part-1-1efc4779568f
15. Hardin, J., Hoerl, R., Horton, N. J., Nolan, D., Baumer, B., Hall-Holt, O., Murrell, P., Peng, R., Roback, P., Temple Lang, D., & Ward, M. D. (2015). Data Science in Statistics Curricula: Preparing Students to "Think with Data." The American Statistician, 69(4), 343–353. https://doi.org/10.1080/00031305.2015.1077729.
16. Harshdeep Gupta (2018) Recommender Systems using Deep Learning in PyTorch from scratch. https://towardsdatascience.com/recommender-systems-using-deep-learning-in-pytorch-from-scratch-f661b8f391d7
17. He, R., & McAuley, J. (2016). Ups and Downs: Modeling the Visual Evolution of Fashion Trends with One-Class Collaborative Filtering. *Proceedings of the 25th International Conference on World Wide Web - WWW '16*, 507–517. https://doi.org/10.1145/2872427.2883037
18. Hopfgartner, F., Lommatzsch, A., Kille, B., Larson, M., Brodt, T., Cremonesi, P., & Karatzoglou, A. (2020.). The Potentials of Recommender Systems Challenges for Student Learning. 3. https://eprints.gla.ac.uk/131314/7/131314.pdf.
19. Huang, K.H. (2018) Introduction to Recommender System. Part 1 (Collaborative Filtering, Singular Value Decomposition). https://medium.com/hackernoon/introduction-to-recommender-system-part-1-collaborative-filtering-singular-value-decomposition-44c9659c5e75
20. Hui, J. (2020) Machine Learning — Recommender System. https://medium.com/@jonathan_hui/machine-learning-recommender-system-e3237b9df14a

21. Jordan, K. (2015). Massive Open Online Course Completion Rates Revisited: Assessment, Length and Attrition. http://www.irrodl.org/index.php/irrodl/article/view/2112/3340
22. Kang, W.-C., Fang, C., Wang, Z., & McAuley, J. (2017). Visually-Aware Fashion Recommendation and Design with Generative Image Models. *2017 IEEE International Conference on Data Mining (ICDM)*, 207–216. https://doi.org/10.1109/ICDM.2017.30
23. Kolb, D. A., Boyatzis, R. E., & Mainemelis, C. (2014). Experiential Learning Theory: Previous Research and New Directions. In R. J. Sternberg & L. Zhang (Eds.), Perspectives on Thinking, Learning, and Cognitive Styles (0 ed., pp. 227–248). Routledge. https://doi.org/10.4324/9781410605986-9
24. Kolb, A. Y., & Kolb, D. A. (2009). The Learning Way: Meta-cognitive Aspects of Experiential Learning. *Simulation & Gaming*, *40*(3), 297–327. https://doi.org/10.1177/1046878108325713
25. Kolb, A. Y., & Kolb, D. A. (2005). Learning Styles and Learning Spaces: Enhancing Experiential Learning in Higher Education. *Academy of Management Learning & Education*, *4*(2), 193–212.
26. Kollwitz (2017). Tools for Academic Business Intelligence & Analytics Teaching—Results of an Evaluation,. Springer International Publishing, p227. https://doi.org/10.1007/978-3-319-58097-5
27. Konstan, J. (2020). Recommender Systems Specialization. https://www.coursera.org/specializations/recommender-systems
28. Kordík, P. (2018). Machine Learning for Recommender systems — Part 1 (algorithms, evaluation and cold start). https://medium.com/recombee-blog/machine-learning-for-recommender-systems-part-1-algorithms-evaluation-and-cold-start-6f696683d0ed
29. Kouki, P., Schaffer, J., Pujara, J., O'Donovan, J., & Getoor, L. (2019). Personalized explanations for hybrid recommender systems. Proceedings of the 24th International Conference on Intelligent User Interfaces - IUI '19, 379–390. https://doi.org/10.1145/3301275.3302306
30. Kross, S., & Guo, P. J. (2019). Practitioners Teaching Data Science in Industry and Academia: Expectations, Workflows, and Challenges. Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems - CHI '19, 1–14. https://doi.org/10.1145/3290605.3300493
31. Lee, H., Seol, J., & Lee, S. (2017). Style2Vec: Representation Learning for Fashion Items from Style Sets. 6. https://arxiv.org/abs/1708.04014

32. Lewis, L. H., & Williams, C. J. (1994). Experiential learning: Past and present. New Directions for Adult and Continuing Education, 1994(62), 5–16. https://doi.org/10.1002/ace.36719946203
33. Li, Y., Cao, L., Zhu, J., & Luo, J. (2017). Mining Fashion Outfit Composition Using an End-to-End Deep Learning Approach on Set Data. *IEEE Transactions on Multimedia*, *19*(8), 1946–1955. https://doi.org/10.1109/TMM.2017.2690144
34. Liu, Z., Luo, P., Qiu, S., Wang, X., & Tang, X. (2016). DeepFashion: Powering Robust Clothes Recognition and Retrieval with Rich Annotations. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1096–1104. https://doi.org/10.1109/CVPR.2016.124
35. Lommatzsch, A., Kille, B., Hopfgartner, F., Larson, M., Brodt, T., Seiler, J., & Özgöbek, Ö. (2017). CLEF 2017 NewsREEL Overview: A Stream-Based Recommender Task for Evaluation and Education. In G. J. F. Jones, S. Lawless, J. Gonzalo, L. Kelly, L. Goeuriot, T. Mandl, L. Cappellato, & N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction (Vol. 10456, pp. 239–254). Springer International Publishing. https://doi.org/10.1007/978-3-319-65813-1_23
36. Moon, J. A. (2013). A Handbook of Reflective and Experiential Learning: Theory and Practice (0 ed.). Routledge. https://doi.org/10.4324/9780203416150
37. Ramamurthy, B. (2016). A Practical and Sustainable Model for Learning and Teaching Data Science. Proceedings of the 47th ACM Technical Symposium on Computing Science Education - SIGCSE '16, 169–174. https://doi.org/10.1145/2839509.2844603.
38. Rocca, B. (2019). Introduction to recommender systems. https://towardsdatascience.com/introduction-to-recommender-systems-6c66cf15ada
39. Saltz, J., & Heckman, R. (2015.). Big Data science education: A case study of a project- focused introductory course. 10. https://pdfs.semanticscholar.org/58b0/bab49633c23e53910925dabd472bab2a99b9.pdf
40. Schuff (2017). Data Science for All: A University-Wide Course in Data Literacy. Springer International Publishing, p281. https://doi.org/10.1007/978-3-319-58097-5.
41. Smith, B., & Linden, G. (2017). Two Decades of Recommender Systems at Amazon.com. IEEE Internet Computing, 21(3), 12–18. https://doi.org/10.1109/MIC.2017.72

42. Swamy, V. (2018.). Pedagogy, Infrastructure, and Analytics for Data Science Education at Scale. 46. http://www2.eecs.berkeley.edu/Pubs/TechRpts/2018/EECS-2018-81.html
43. Towards Data Science (2020). https://towardsdatascience.com/
44. Udemy, (2020). Recommendation Engine Courses https://www.udemy.com/topic/recommendation-engine/
45. Xiao, H., Rasul, K., & Vollgraf, R. (2017). Fashion-MNIST: A Novel Image Dataset for Benchmarking Machine Learning Algorithms. ArXiv:1708.07747 [Cs, Stat]. http://arxiv.org/abs/1708.07747
46. Zhang, S., Yao, L., Sun, A., & Tay, Y. (2019). Deep Learning based Recommender System: A Survey and New Perspectives. ACM Computing Surveys, 52(1), 1–38. https://doi.org/10.1145/3285029
47. Zheng, Y. (2019). A Course on Applied AI and Data Science: Recommender Systems. *Proceedings of the 20th Annual SIG Conference on Information Technology Education - SIGITE '19*, 43–48. https://doi.org/10.1145/3349266.3351405